LOAN DOCUMENT

PHOTOGRAPH THIS SHEET

DTIC ACCESSION NUMBER

LEVEL

0

INVENTORY

Lessons Learned from the First Capability...

DOCUMENT IDENTIFICATION

Sep 96

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

H A N D L E W I T H C A R E

ACCESSION FOR

NTIS	GRA&I	☑
DTIC	TRAC	☐
UNANNOUNCED		☐
JUSTIFICATION		

BY

DISTRIBUTION/

AVAILABILITY CODES

DISTRIBUTION	AVAILABILITY AND/OR SPECIAL	
A-1		

DISTRIBUTION STAMP

DATE ACCESSIONED

19970114 009

DATE RETURNED

DATE RECEIVED IN DTIC

REGISTERED OR CERTIFIED NUMBER

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-FDAC

DTIC QUALITY INSPECTED 1

17 JAN 1997

LESSONS LEARNED FROM THE FIRST CAPABILITY INCREMENT OF THE NATIONAL MISSILE DEFENSE (NMD) BATTLE MANAGEMENT / COMMAND, CONTROL, AND COMMUNICATIONS (BMC3) SOFTWARE

Captain Jeff Blank
ESC/XRS
Hanscom AFB, Bedford MA 01731

Mary L. Urban
Charles K. Wilkinson
The MITRE Corporation
Bedford, MA 01730

Abstract

A demonstrator system for the Battle Management, Command, Control, and Communications element of the National Missile Defense system is being built in seven increments. This paper reports lessons learned from development of the first increment. Four lessons are discussed. First, a relatively informal requirements baseline, generated and iterated by the contractor, was found to meet the needs of the program. Second, benefits from use of object-oriented methods and Ada95 will not be realized until later increments. Third, there were successful alternatives to the reviews and documents eliminated in acquisition streamlining. Lastly, vigilance to keep process versus product emphasis in balance was needed.

Background

The aim of the National Missile Defense (NMD) program is to develop a system of systems with the capability to defend the nation from the threat of limited ballistic missile attacks. The decision to deploy the system will be made in FY99 or later based on the emergence of a rogue nation threat. Once a deployment decision is made, the program is structured to field an initial operational capability within three years. The Battle Management, Command, Control, and Communications (BMC3) system is a key element of the overall NMD program. Its role is to tie together various ground- and space-based sensors with the ground-based interceptor to give the warfighter the ability to defeat incoming attacks. This paper addresses lessons learned from development of the first incremental release of the BMC3 software.

The NMD program is managed by the Ballistic Missile Defense Organization (BMDO), in cooperation with the military services (Army, Air Force, Navy) as executing agents for most of the elements. The Ground-Based Interceptor (GBI) is a non-nuclear, hit-to-kill weapon being developed by the Army. The Army is also managing development of the Ground-Based Radar (GBR), which performs detailed tracking and threat discrimination in X-band to provide fire-control support to the GBI. The Air Force is developing the Space and Missile Tracking System (SMTS), a space-based sensor with the capability of tracking and discriminating threat objects through their post-boost and mid-course phases. The Air Force is also developing the Upgraded Early Warning Radar (UEWR), which, as its name implies, is an upgrade to existing radars to perform early to midcourse threat tracking and discrimination. The UEWRs will provide cueing data to the GBR to facilitate search and detection. It is needed until the SMTS is fully operational. The system will also take advantage of existing satellites to detect launches. The BMC3 element discussed in this paper is unique in that BMDO retains overall program management authority while using the Air Force and Army as executing agents for technical direction and management oversight of separate portions of the development effort. The Navy is participating in the NMD program by conducting Independent Verification and Validation (IV&V) of the BMC3 software. A diagram of the complete NMD system is shown in Figure 1.

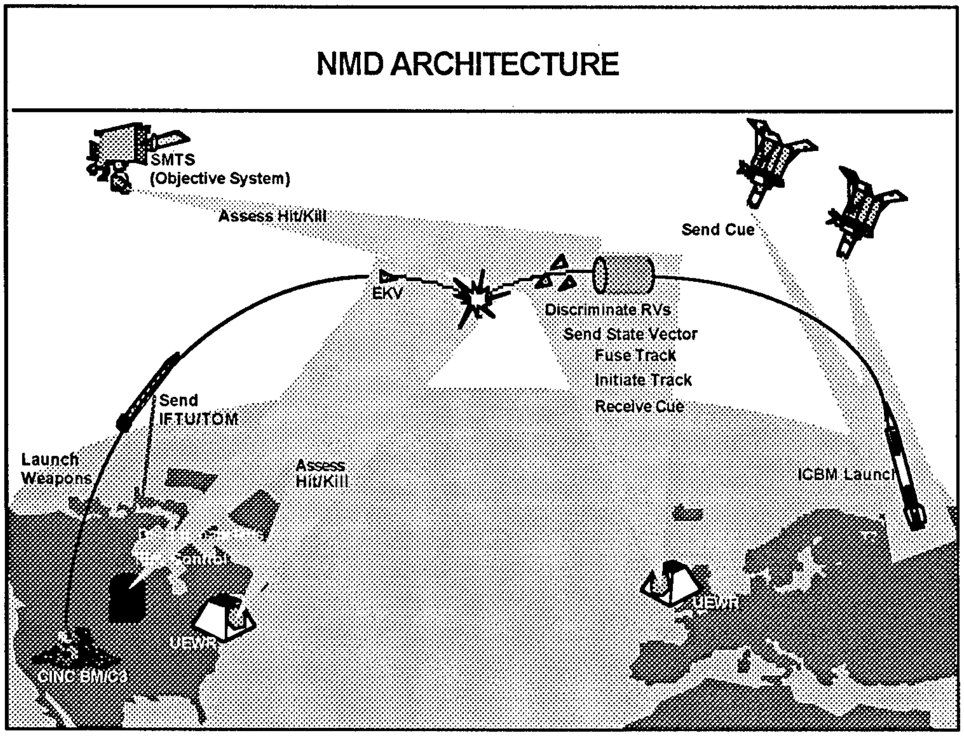

Figure 1. NMD Architecture

BMC3 System Description

The BMC3 system is used to manage the sensor, weapon, and communications resources in accomplishing the NMD mission. It is the integrating element within the NMD system of elements and it integrates the NMD system with external systems that support the missile negation mission. It provides the means to plan for, assess, direct, coordinate, monitor, and control all aspects of the missile defense battle.

The Command and Control (C2) function is the exercise of authority and direction over assigned resources in the accomplishment of the mission. The essence of Command and Control is to take charge when the defense design fails. This is extremely challenging since it may not be obvious that the defense design is failing, and decisions must be made in real time. Decision support and flexibility in the controls on the automated algorithms provide for the command and control needs. The Battle Management (BM) function is the set of automated operations that process surveillance data, respond to C2 system control directives, and perform planning required to task the supporting sensor, weapon, and communications resources. Communications provides the hardware and software resources to securely send and receive information/data not only within the BMC3 element, but between the BMC3 element and other NMD elements and supporting external systems.

To accomplish a defense, the BMC3 system will receive and process missile launch and track data in support of situation assessment and will determine if Rules of Engagement are fulfilled. Cues will be provided to midcourse tracking sensors to minimize sensor search requirements. Authorized users will be given the ability to select overall strategies and tactics, allocate and authorize use of resources, and issue and disseminate the associated directives. Sensor data will be correlated and fused to maintain a consistent perception of the battle situation. Coordinated task plans will be developed and issued to sensor, weapon, and communication resources. In-flight target updates and target object maps will be generated and sent to the kinetic kill vehicles to enhance the probability of target kills. The progress of the battle will be monitored and evaluated to support potential updates to strategies and tactics.

During peacetime BMC3 will: develop and update battle plans; conduct and support tests, training, and exercises; conduct maintenance; monitor the health and status of supporting systems; and support collateral missions.

CI-1 Product Description

The CI-1 product is described in terms of its Integrated Engineering Infrastructure (IEI) and its Mission Applications Software (MAS). The IEI is the set of hardware and software used for the development and implementation of the CI and is depicted in Figure 2. The intended runtime configuration for CI-1 consists of one or two hardware/software suites representing BMC3 nodes. Each node has six Silicon Graphics Indigo Workstations. Key products in the runtime environment are: the Silicon Graphics IRIX operating system; Universal Network Applications Services (UNAS), a software product consisting of a suite of portable reusable components and services for developing distributed heterogeneous, message-based systems; Sybase, a data base management system; and TeleUSE to support an X Windows/MOTIF user interface. Mission specific application software provides limited functionality to represent a nominal engagement scenario and perform planning and tasking to counter that threat. Missile launch and track data is received from NMD and external systems. This information is correlated and fused to create and maintain a system threat picture. Displays consist of a 3-D globe and information displays. Health and status of supporting systems is monitored, displayed, and used in planning the engagement. Coordinated task plans for the weapons, sensors, and communication are developed based on a selected engagement strategy. These plans are scheduled and issued to the appropriate interfaces.

Development Approach

The BMC3 software development program was structured from the beginning to implement established best practices and emerging acquisition streamlining efforts. An incremental development approach is being used, with versions of the software released approximately every year of the planned seven year effort. This approach allows meaningful user evaluation and feedback throughout development and mitigates the effects of changing requirements over the life of the program. Information architectures (IAs) are being used to model the structure of the overall NMD system and the BMC3 element. These models directly feed into the coding effort. The software itself is largely written in Ada95, an object-oriented language. This facilitates reuse between increments and provides flexibility to accommodate changes in the system, and extensions to the capabilities. The development effort is managed cooperatively among the various players using integrated product teams with few traditional data deliveries or formal reviews.

The BMC3 development effort is a major portion of a contract awarded to TRW Strategic Systems Division in August 1995 by BMDO. TRW is conducting the BMC3 development program from three major locations: 1) System and BMC3 element-level modeling and requirements definition are being conducted at TRW's Rosslyn, VA facility. 2) Command and Control (C2) and Integrated Engineering Infrastructure (IEI) development are being conducted at TRW's facility in BMDO's Joint National Test Facility (JNTF) at Falcon AFB, CO. 3) The remainder of the development effort, including development of the communications, test exerciser, and engagement planning is being conducted at TRW's Huntsville, AL, facility.

BMDO assigned responsibility for leading development of the first incremental BMC3 software release to the Air Force, specifically the NMD BMC3 program office in the Developmental Planning Directorate of the Electronic Systems Center located at Hanscom Air Force Base, MA. Members of the Capability Increment One (CI-1) "Build Team" included representatives from MITRE Corporation, SenCom Corporation, the Army's BMC3 program office, the Navy IV&V lead, a user representative from Air Force Space Command, a BMDO representative, TRW representatives from both BMC3 development sites in Huntsville, AL and Colorado Springs, CO, and various support contractors. Early in the nine-month development of CI-1, the Build Team adopted the following list of priorities to guide their actions:

1. Develop a flexible BMC3 infrastructure and architecture
2. Meet the requirements that were baselined at the CI-1 initial review (IR)

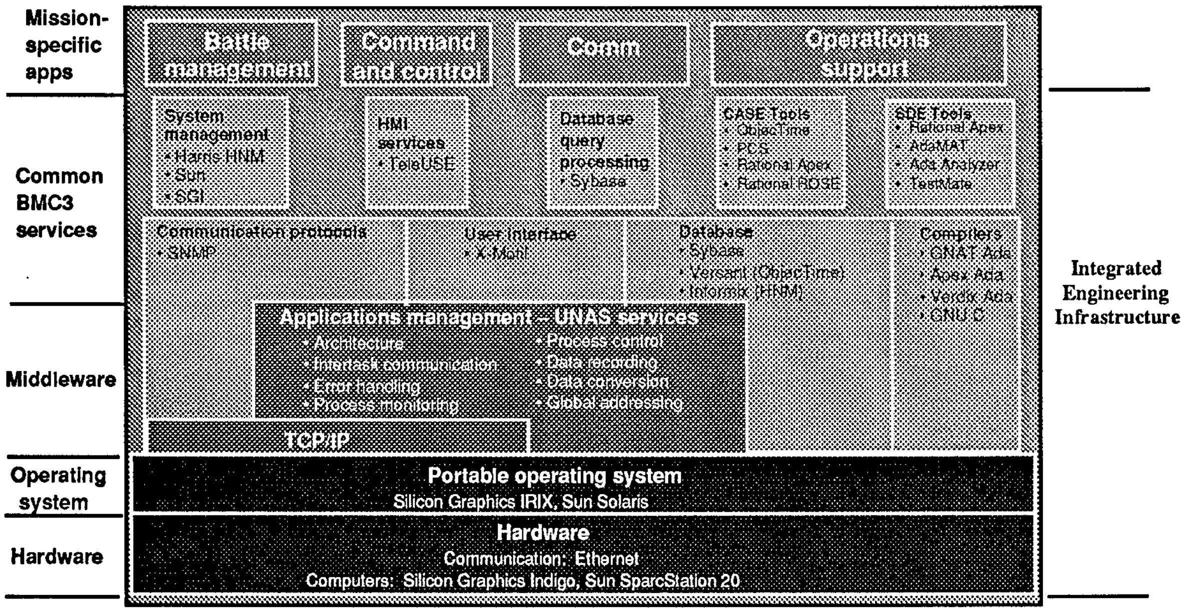

Figure 2. CI-1 Product Components

3. Deliver the 90% solution on-time and on-budget
4. Provide an initial BMC3 capability to leverage early In Flight Tests (IFTs)
5. Demonstrate an effective team approach and process for product development
6. Provide a means to transfer products and lessons learned to Capability Increment 2 (CI-2)

A brief history of the CI-1 development effort will help put the lessons learned into context. The Initial Review (IR) was conducted in February and March 1996, with the intent of establishing the requirements to which the CI would be built. Early in the project, several weeks of schedule were lost establishing the Ada95 development environment and recompiling Commercial Off-The-Shelf (COTS) software. In early April, the entire NMD program transitioned from a technology readiness effort to a deployment readiness program amid growing congressional interest. While not directly affecting CI-1 development since the baselined requirements were not altered, this event did change expectations for the overall development effort.

In late April, BMDO decided to use CI-1 to support an integrated flight test (IFT-1) scheduled to occur relatively soon after this increment was due to be released. This increased the project's schedule pressure, increased the development and integration work to be done, and defocused the project's attention somewhat from the original intent of the build: developing a flexible BMC3 architecture. The project manager decided at this time to defer to future increments all work on decision aids (a major portion of the command and control capability) in order to increase the likelihood of meeting schedule

Throughout May, June, and July, the contractor made excellent progress in the design and code portions of the development effort, although staying approximately one month behind schedule due to the early problems. In July, the government project manager announced a one month slip of Release Review (RR) to reflect the behind schedule condition. At the same time, some relatively minor changes to the requirements were made. As of this writing, release testing is scheduled for early- and mid-September to ensure that the CI-1 product meets the baselined requirements and the product is due to be released at the end of September. The Lessons Learned documented in the remainder of this paper fall under four topics: 1) requirements, 2) use of object-oriented methods and Ada95, 3) acquisition streamlining, and the 4) balance between process and product.

Lesson Learned #1

> The requirements baseline generated and iterated by the contractor, using an ad hoc process and based on a relatively informal definition of the CI-1 content, was satisfactory.

The NMD program has a long history with many changes in scope and architecture. A number of out-of-date requirements documents, both System Requirements Documents and Element Requirements Documents, have been developed in previous efforts. Familiarity with these products meant that the Government and contractor personnel knew and understood the functionality which would be required of the Objective system. Because the BMC3 system engineering and CI-1 software development efforts were concurrent in the BMC3 program, the requirements for CI-1 had to be baselined far before specifications were developed for the Objective BMC3 element. In fact, the contractor was actually far into the design of this first increment prior to the CI-1 requirements being officially baselined by the Government. The software developers used the IR viewgraph presentation as the starting point for the requirements. They translated that definition of the CI-1 content into a set of about 100 trackable and testable statements. Figure 3 illustrates both the "as bid" and the actual requirements processes. While there were major changes to the list of requirements throughout the seven months between the initial and release reviews, the content of CI-1 did not change substantially, with the exception of the deferment of the decision aids effort. We believe that the experience and domain knowledge of the software development team and the existence of legacy requirements documents for NMD and BMC3 were factors that contributed to successful interpretation of the requirements.

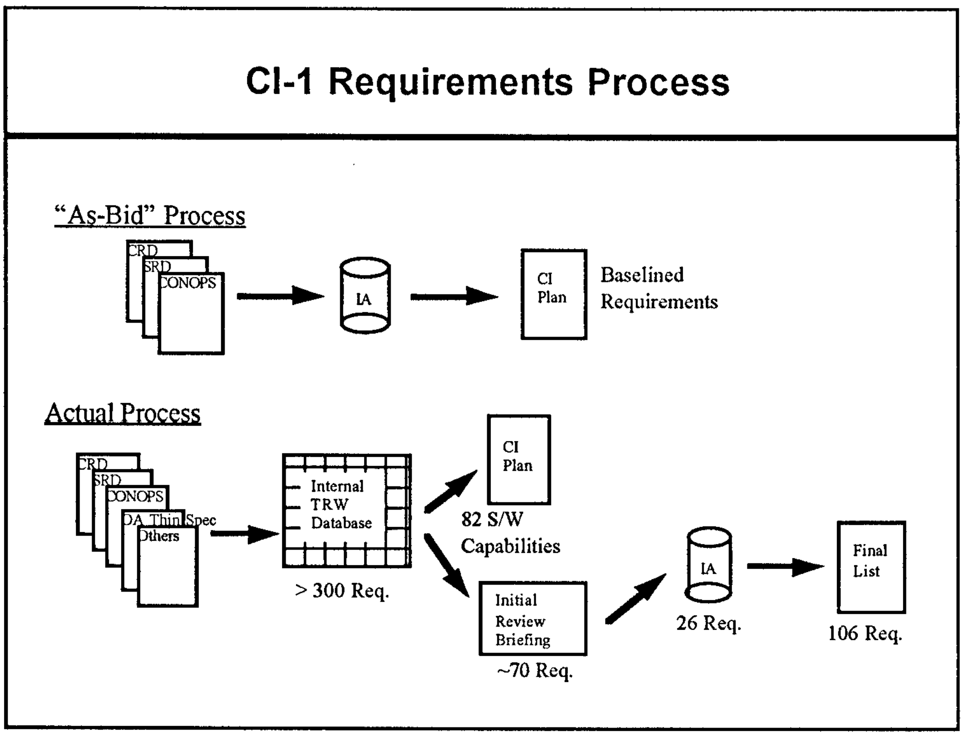

Figure 3. CI-1 Requirements Process

These statements have satisfied those who need to use a requirements baseline. They are accepted as the agreement between BMDO and the Build Team on what will be built in the CI. They are the management tool used to track recent and potential schedule/content trades. The statements establish the requirements for release testing that will establish that the product is ready for user assessment and participation in NMD system tests, including integrated NMD ground and flight tests. They also have been fed into the system engineering activities as a bottoms-up input to the Objective BMC3 requirements definition.

There was concern that there might be unmet expectations from the broader BMC3 community regarding the content of this first increment. The community, more specifically the user, never reviewed a specification for CI-1, but did have impressions and inherent expectations from other prototypes and Wargames. The Government project manager briefed the intended content at many project and program reviews with the intent of establishing community expectations. Routine and engagement operations supported by the CI, annotated for requirements changes, are shown in Figures 4 and 5. Over time, the character of the increment and the

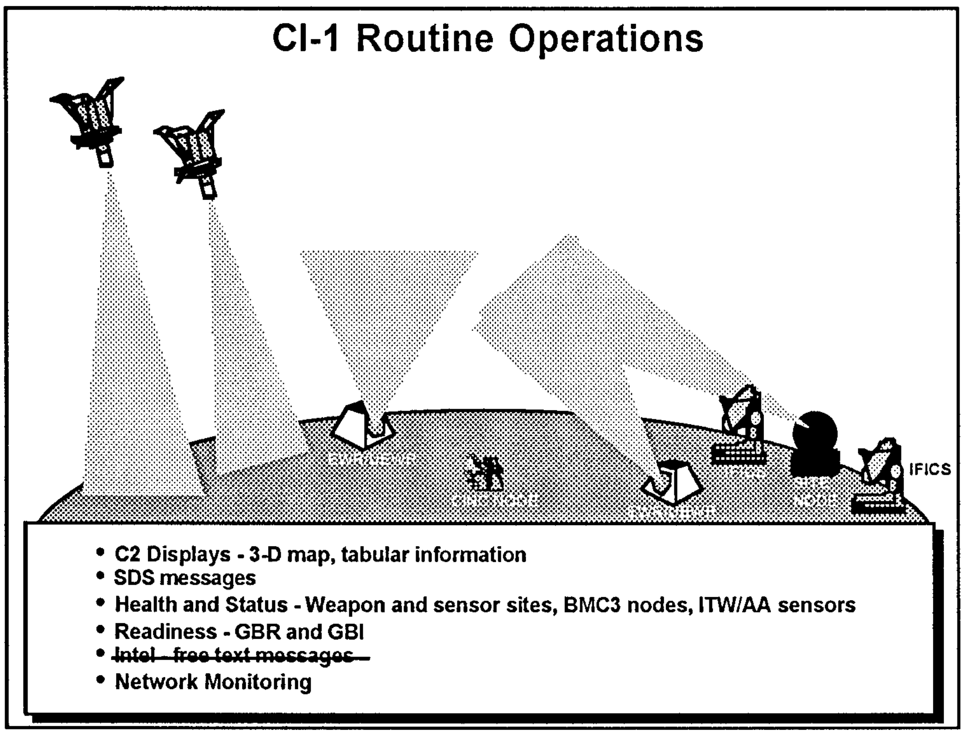

Figure 4. CI-1 Routine Operations

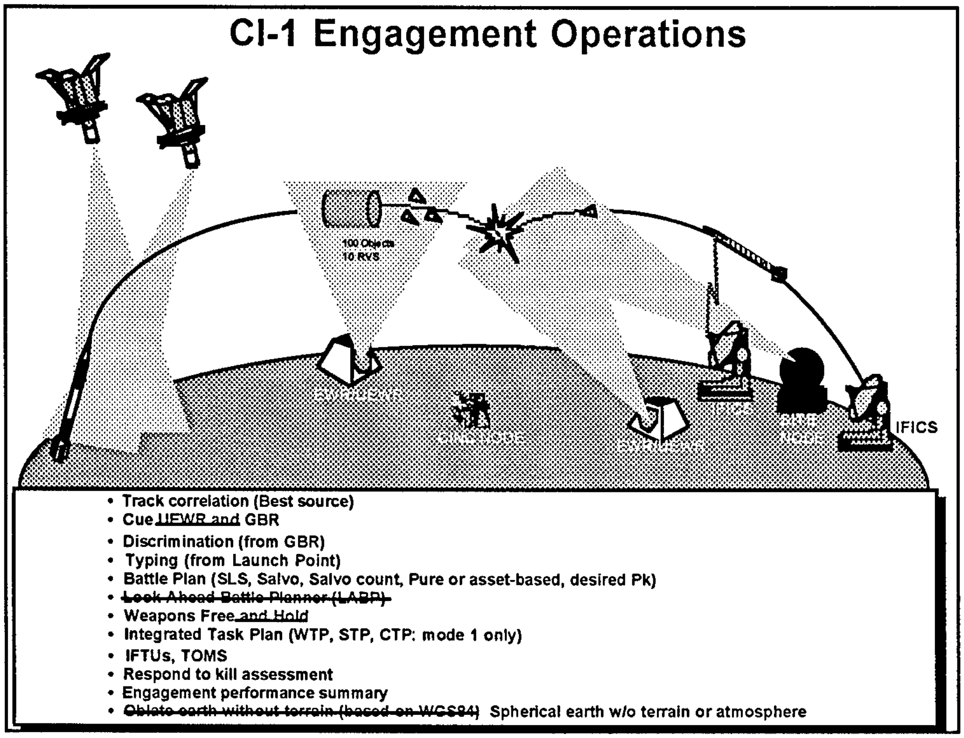

Figure 5. CI-1 Engagement Operations

need for an architecture increment with limited functionality seemed to be accepted. This substituted for the buy-in that usually occurs through requirement document comment resolution meetings. One problem with the requirements baseline that was established for CI-1 was the strong focus on mission applications, to the exclusion of software architecture or IEI issues. This is understandable, in that mission applications capture the attention of the community and trace directly to performance specifications. However, the mission applications depend on the software architecture and the IEI. Since this was intended to be the "architecture build", the requirements should have addressed such issues as flexibility, evolvability, and extensibility of the software architecture and the IEI.

Lesson Learned #2

Anticipated benefits from use of object-oriented methods and Ada95 will not come until later increments.

The contractor (with strong encouragement from the Government) is using object-oriented methods for both system engineering and software development. The BMC3 Information Architecture is an essential part of the contractors engineering approach. It is being developed in the Real-Time Object-Oriented Modeling (ROOM) methodology using the ObjecTime CASE tool. Software design is accomplished using another methodology, Object-Modeling Technique (OMT) and the Rational ROSE tool, as well as ROOM and ObjecTime. New code is being written in Ada95 and compiled with the GNU New York University Ada Translator (GNAT) compiler. The objective of this, the "architecture increment", is to construct a stable and complete object structure for the mission application software that can be fleshed out to implement additional capability in future increments.

We anticipate benefits of modularity, flexibility, maintainability, evolvability, and extensibility from this approach. Specifically, automated tools will produce code from the IA, metrics will be automatically generated that indicate the quality and quantity of code, software from one increment can easily be reused in the next increment, and the impact of requirements changes on the software will be confined by the structure of the objects.

ROOM constructs, e.g., actor classes, protocol classes and data classes, represent software entities in an OO model and correspondingly, in the Software Architecture Skeleton (SAS) for the mission applications software. At this time, the SAS is expected to be the basis of software development for CI-2, and subsequent increments as well, with new work confined to subclasses and methods. The emphasis on this internal structure is the reason that the more visible aspects of the product are so limited. It is more important, for example, to abstract the common elements of sensor and weapon tasking, to achieve a good object-oriented implementation, than to implement a more sophisticated user interface. These priorities, imposed by the OO paradigm, can be erroneously perceived by many as indcations of slow progress.

There were other factors that impacted the rate of progress. The first is the lack of availability of an integrated tool set for OO and Ada95. The second is the scarcity of trained programmers. The OO tools available typically support C++ development. Ada95 is a new language with new compilers. The contractor used the GNAT compiler and partially integrated it with some of the other engineering tools but the limitations of the tool set slowed progress. From a management perspective, OO metrics were to be collected and reported monthly to the Government. There were no automated tools for collecting the OO metrics on Ada95 also required that the UNAS, the COTS software product used for interprocess communication, had to be recompiled in Ada95. This imposed a specific and real delay which contributed to a one month slip in the release of the increment.

Many of the contract personnel were experienced in the domain of command and control for missile defense. There were also object-oriented methodologists. The number of people with experience in both BMC3 and object-oriented methods was significantly fewer. Experience with Ada95 was unavoidably lacking and training in the new language was required.

Flexibility is required for the BMC3 system. The goal of building in features that support flexibility is to minimize the impact of change. For BMC3, change can mean new or different weapons and sensors, new BMC3 nodes, new command and control structures. The certainty of these changes is a lesson learned from the many aborted requirements efforts noted earlier. The developer has implemented several features, facilitated by the OO aspects of the design to achieve this. Each BMC3 node will have identical software. There is a system of tokens, assigned to objects, to indicate which node has the active processing and which is in the background ready and able to assume the primary role. Another feature is the abstraction applied for processing and data relative to different types of sensors.

At this time, we can only say that we are satisfied with our product. The features of the product are the SAS with only limited functionality and limited user actions. We still believe that this was the correct approach for the first increment.

Lesson Learned #3

Oversight and insight require alternatives to the reviews and documents eliminated in acquisition streamlining.

The BMC3/SE&I contract included an Integrated Management Plan (IMP) and an Integrated Master Schedule (IMS) written by the contractor. These documents were to be used to guide the work effort instead of the traditional processes and products imposed by military standards. The IMP and IMS both were soon out of date. They are both substantial, detailed documents and contract modifications have not kept up with the many changes. They have not been used as management tools on the CI-1 development.

Plans for a Contact Management Information System (CMIS) to connect all program participants have yet to be fully accomplished, for a variety of reasons not attributable to any one organization. This was to be the means by which everyone had access to the contractor's engineering data. There is essentially no requirement or design documentation required from or produced by the contractor. Rather, the contractor's Information Architecture and software design descriptive material reside in on-line databases.

Government oversight and technical insight was accomplished through weekly teleconferences, monthly Build Team meetings, a series of software pacing benchmarks, design walkthroughs (DWTs) and code walk through (CWTs) as shown in Figure 6. Preliminary Design Review, Critical Design Review, Software Requirements Specifications, and Design documentation were not required.

For the most part, the alternatives worked as reasonable substitutes for reviews of the mission application software. The contractor is an experienced and successful developer of systems similar to BMC3 and as such, has applied processes and system concepts that were successful on previous projects. Formal activities to establish compliance with requirements were made unnecessary by the informal nature of the requirements. The Pacing Benchmarks addressed incremental content in the software and identified five points at which capabilities could be assessed to determine if the scheduled progress was being achieved. The first pacing benchmark focused attention on the UNAS recompile, at the second the C2 displays were constructed, and the third and fourth measured the integration of software developed in the two locations, Huntsville and Colorado Springs. The last pacing benchmark was critical, as it exercised distributed processing between two locations, representing two BMC3 nodes. These benchmarks were more meaningful milestones in our environment than numbers of requirements implemented or modules designed would have been.

Design and code walkthroughs, primarily software developers peer reviews with Government observers, were conducted for the purpose of defects removal. A side benefit to this activity was that it served as a forum for joint Government/contractor technical sessions. These were really the only structured opportunity for the Government to have insight into the design. Documentation was generated prior to the walkthrough and reflected a description of the design or code at a single point in time, not necessarily the same time for all. This is all that was available for technical documentation. Continuous access to the database or regular deliveries of documentation are necessary to effectively provide technical oversight of the project. As an alternative, MITRE is making an effort to document the architecture and design, then review it for correctness with the contractor. It is hoped that this will have minimal impact on the contractor's schedule, but will provide what is perceived by the Government as necessary. We are assuming that the contractor's databases will suffice for maintainability but have to rely on the contractor to ensure that.

Lesson Learned #4

> Vigilance to keep process versus product emphasis in balance is needed.

The right product is the goal of all involved. Process helps the government in assessing the value of the product and the contractor in controlling the development. The RFP for BMC3/SE&I emphasized process. It required the contractor to write an Integrated Management Plan to define the processes to be used throughout the contract. There was no specification of the product either provided by the Government or required with the proposal. Separate contracts and Government efforts have applied substantial resources on this program to investigate use of Information Architecture, requirements analysis methods, use of various OO methodologies, and approaches to evolutionary development. There is a long range view of the product, a deployment decision is not expected for several years, and the product is intended to be on the path towards an operational capability.. From a different perspective, the evolving BMC3 prototype is to operate in system tests, including flight tests, and annual deliveries are to be assessed by the user. Both the operational evolvability and flight test support put demands on the product.

The CI-1 Build Team set explicit priorities when it formed. One of these was to give process and product equal emphasis. It was impossible to assign greater weight to either. The process would be the precedent for all future increments. The product would be the foundation for all future increments. When the requirements process bogged down, and requirements were being reworked to respond to criticisms that they were not following the prescribed methodology, work on the product did not stop. Software modules that the developer knew were going to be needed were being coded before the requirements were set. A short term approach to integrating Ada95 code with UNAS was implemented when the schedule was in jeopardy.

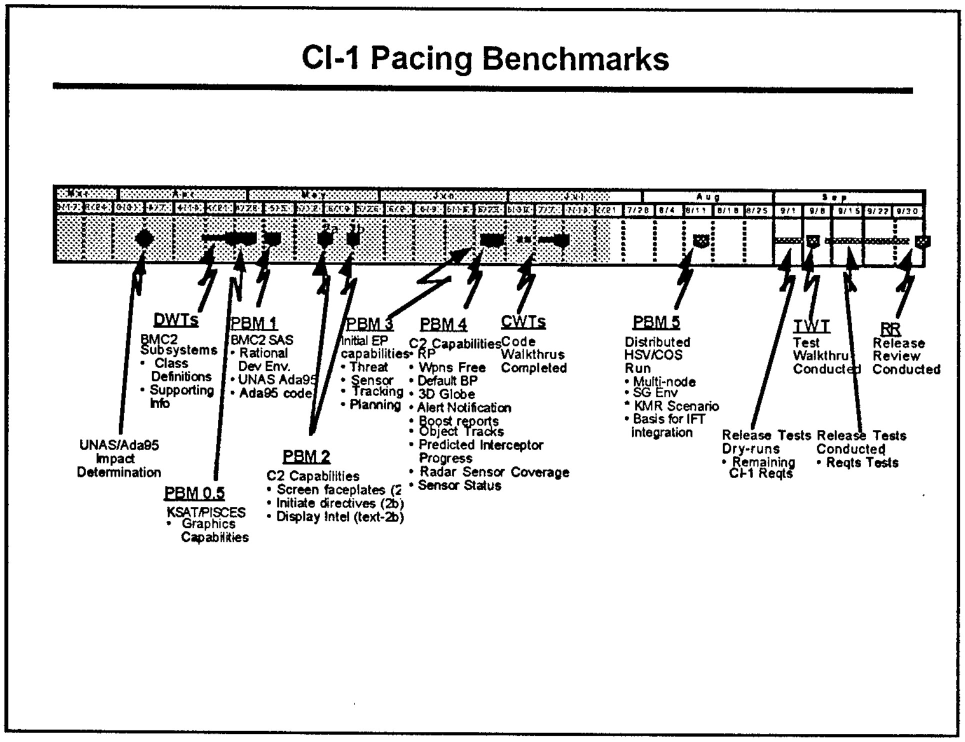

Figure 6. CI-1 Pacing Benchmarks and Walkthroughs

Interest from the BMC3 community was more on the new process concepts like Information Architecture and OO methodologies rather than on familiar mission application requirements. The CI-1 product consists of an Integrated Engineering Infrastructure, a Software Architecture Skeleton for mission applications, and software to support the basic peacetime and engagement capabilities required of it.

Conclusions

The goals of the CI-1 development effort are being met. We have developed a robust BMC3 architectural skeleton with the ability to accommodate a variety of NMD system architectures and increasing BMC3 mission applications, while at the same time meeting almost all of the baselined mission application requirements for this increment. We have instituted software development processes, including OO methods and an Ada95 development environment, which appear to have been successful in their application to this increment and hopefully will serve as the basis for future increments. We have established management processes and structures which increase communication and participation by all parties through the use of streamlined acquisition principles. We look forward to participating in further development of the BMC3 system as it builds on this established foundation.